

CLAIMS

What is claimed is:

- 1 1. An apparatus comprising:
2 a processor executive (PE) to handle an operating system executive (OSE) in a
3 secure environment, the secure environment having a platform key (PK) and associated
4 with an isolated memory area in a platform, the OSE to manage a subset of an operating
5 system (OS) running on the platform, the platform having a processor operating in one
6 of a normal execution mode and an isolated execution mode, the isolated memory area
7 being accessible to the processor in the isolated execution mode;
8 a PE supplement to supplement the PE with a PE manifest representing the PE
9 and a PE identifier to identify the PE; and
10 a PE handler to handle the PE using the PK and the PE supplement.
- 1 2. The apparatus of claim 1 further comprises:
2 a boot-up code to boot up the platform following a power on.
- 1 3. The apparatus of claim 2 wherein the secure environment includes an
2 OSE supplement to supplement the OSE with an OSE manifest representing the OSE
3 and an OSE identifier to identify the OSE.
- 1 4. The apparatus of claim 3 wherein the PE handler comprises:
2 a PE loader to load the PE and the PE supplement from a PE memory into the
3 isolated memory area using a parameter block provided by the boot-up code;
4 a PE manifest verifier to verify the PE manifest; and
5 a PE verifier to verify the PE using the PE manifest and a constant derived from
6 the PK.
- 1 5. The apparatus of claim 4 wherein the PE handler further comprises:
2 a PE key generator to generate a PE key using the PK;
3 a PE identifier logger to log the PE identifier in a storage; and
4 a PE entrance/exit handler to handle a PE entry and a PE exit.

002260" 092200

1 6. The apparatus of claim 5 wherein the PE key generator comprises:
2 a PE key combiner to combine the PE identifier and the PK, the combined PE
3 identifier and the PK corresponding to the PE key.

1 7. The apparatus of claim 6 wherein the PE comprises:
2 an OSE loader to load the OSE and the OSE supplement into the isolated
3 memory area;
4 an OSE manifest verifier to verify the OSE manifest; and
5 an OSE verifier to verify the OSE.

1 8. The apparatus of claim 7 wherein the PE further comprises:
2 an OSE key generator to generate an OSE key;
3 an OSE identifier logger to log the OSE identifier in a storage; and
4 an OSE entrance/exit handler to handle an OSE entry and an OSE exit.

1 9. The apparatus of claim 8 wherein the OSE key generator comprises:
2 a binding key generator to generate a binding key (BK) using the PE key; and
3 an OSE key combiner to combine the OSE identifier and the BK, the combined
4 OSE identifier and the BK corresponding to the OSE key.

1 10. The apparatus of claim 9 wherein the OSE comprises:
2 a module loader to load a module into the isolated memory area;
3 a page manager to manage paging in the isolated memory area; and
4 an interface handler to handle interface with the OS.

1 11. The apparatus of claim 9 wherein the module is one of an application
2 module, an applet module, and a support module.

1 12. The apparatus of claim 11 wherein the OSE further comprises:
2 an applet key generator to generate an applet key associating with the applet
3 module.

1 13. The apparatus of claim 12 wherein the applet key generator comprises:
2 an applet key combiner to combine the OSE key with an applet identifier
3 identifying the applet module, the combined OSE key and the applet identifier
4 corresponding to the applet key.

1 14. The apparatus of claim 13 wherein the boot up code comprises:
2 a PE locator to locate the PE and the PE supplement, the PE locator transferring
3 the PE and the PE supplement into the PE memory at a PE address;
4 a PE recorder to record the PE address in the parameter block; and
5 an instruction invoker to execute an isolated create instruction, the isolated
6 create instruction loading the PE handler into the isolated memory area.

1 15. The apparatus of claim 14 wherein the isolated create instruction
2 performs an atomic sequence, the atomic sequence being non-interruptible.

1 16. The apparatus of claim 15 wherein the atomic sequence comprises:
2 a physical memory operation to verify if the processor is in a flat physical page
3 mode;
4 an atomic read-and-increment operation to read and increment a thread count
5 register in a chipset, the read-and-increment operation determining if the processor is
6 the first processor in the isolated execution mode;
7 an isolated memory area control operation to configure the chipset using a
8 configuration storage;
9 a processor isolated execution operation to configure the processor in the
10 isolated execution mode; and
11 an PE handler loading operation to load the PE handler into the isolated memory
12 area.

1 17. The apparatus of claim 16 wherein the atomic sequence further
2 comprises:
3 a PE handler verification to verify the loaded PE handler; and
4 an exit operation to transfer control to the loaded PE handler.

1 18. The apparatus of claim 16 wherein the processor isolated execution
2 operation comprises:

3 a chipset read operation to read the configuration storage in the chipset when the
4 processor is not a first processor in the isolated execution mode; and

5 a processor configuration operation to configure the processor according to the
6 configuration storage in the chipset when the processor is not the first processor in the
7 isolated execution mode.

1 19. The apparatus of claim 18 wherein the chipset includes at least one of a
2 memory controller hub (MCH) and an input/output controller hub (ICH).

1 20. The apparatus of claim 8 wherein the storage is in an input/output
2 controller hub (ICH) external to the processor.

1 21. A method comprising:

2 handling an operating system executive (OSE) by a processor executive (PE) in
3 a secure environment, the secure environment having a platform key (PK) and
4 associated with an isolated memory area in a platform, the OSE to manage a subset of
5 an operating system (OS) running on the platform, the platform having a processor
6 operating in one of a normal execution mode and an isolated execution mode, the
7 isolated memory area being accessible to the processor in the isolated execution mode;

8 supplementing the PE using a PE supplement, the PE supplement having a PE
9 manifest representing the PE and a PE identifier to identify the PE; and

10 handling the PE by a PE handler using the PK and the PE supplement.

1 22. The method of claim 21 further comprises:

2 booting up the platform by a boot-up code following a power on.

1 23. The method of claim 22 wherein the secure environment includes an
2 OSE supplement to supplement the OSE with an OSE manifest representing the OSE
3 and an OSE identifier to identify the OSE.

0022260" 58589960

1 24. The method of claim 23 wherein handling the PE comprises:
2 loading the PE and the PE supplement from a PE memory into the isolated
3 memory area using a parameter block provided by the boot-up code;
4 verifying the PE manifest; and
5 verifying the PE using the PE manifest and a constant derived from the PK.

1 25. The method of claim 24 wherein handling the PE further comprises:
2 generating a PE key using the PK;
3 logging the PE identifier in a storage; and
4 handling a PE entry and a PE exit.

1 26. The method of claim 25 wherein generating the PE key comprises:
2 combining the PE identifier and the PK, the combined PE identifier and the PK
3 corresponding to the PE key.

1 27. The method of claim 26 wherein handling the OSE comprises:
2 loading the OSE and the OSE supplement into the isolated memory area;
3 verifying the OSE manifest; and
4 verifying the OSE.

1 28. The method of claim 27 wherein handling the OSE further comprises:
2 generating an OSE key;
3 logging the OSE identifier in a storage; and
4 handling an OSE entry and an OSE exit.

1 29. The method of claim 28 wherein generating the OSE key comprises:
2 generating a binding key (BK) using the PE key; and
3 combining the OSE identifier and the BK, the combined OSE identifier and the
4 BK corresponding to the OSE key.

1 30. The method of claim 29 wherein managing the subset of the OS
2 comprises:

3 loading a module into the isolated memory area;
4 managing paging in the isolated memory area; and
5 handling interface with the OS.

1 31. The method of claim 29 wherein the module is one of an application
2 module, an applet module, and a support module.

1 32. The method of claim 31 wherein managing the subset of the OS further
2 comprises:
3 generating an applet key associating with the applet module.

1 33. The method of claim 32 wherein generating the applet key comprises:
2 combining the OSE key with an applet identifier identifying the applet module,
3 the combined OSE key and the applet identifier corresponding to the applet key.

1 34. The method of claim 33 wherein booting up comprises:
2 locating the PE and the PE supplement;
3 transferring the PE and the PE supplement into the PE memory at a PE address;
4 recording the PE address in the parameter block; and
5 executing an isolated create instruction, the isolated create instruction loading
6 the PE handler into the isolated memory area.

1 35. The method of claim 34 wherein executing the isolated create instruction
2 comprises performing an atomic sequence, the atomic sequence being non-interruptible.

1 36. The method of claim 35 wherein performing the atomic sequence
2 comprises:
3 verifying if the processor is in a flat physical page mode;
4 reading and incrementing a thread count register in a chipset to determine if the
5 processor is the first processor in the isolated execution mode;
6 configuring the chipset using a configuration storage;
7 configuring the processor in the isolated execution mode; and
8 loading the PE handler into the isolated memory area.

1 37. The method of claim 36 wherein performing the atomic sequence further
2 comprises:
3 verifying the loaded PE handler; and
4 transferring control to the loaded PE handler.

1 38. The method of claim 36 wherein configuring the processor in the
2 isolated execution mode comprises:
3 reading the configuration storage in the chipset when the processor is not a first
4 processor in the isolated execution mode; and
5 configuring the processor according to the configuration storage in the chipset
6 when the processor is not the first processor in the isolated execution mode.

1 39. The method of claim 38 wherein the chipset includes at least one of a
2 memory controller hub (MCH) and an input/output controller hub (ICH).

1 40. The method of claim 28 wherein the storage is in an input/output
2 controller hub (ICH) external to the processor.

1 41. A computer program product comprising:
2 a machine useable medium having computer program code embedded therein,
3 the computer program product having:
4 computer readable program code for handling an operating system
5 executive (OSE) by a processor executive (PE) in a secure environment, the
6 secure environment having a platform key (PK) and associated with an isolated
7 memory area in a platform, the OSE to manage a subset of an operating system
8 (OS) running on the platform, the platform having a processor operating in one
9 of a normal execution mode and an isolated execution mode, the isolated
10 memory area being accessible to the processor in the isolated execution mode;
11 computer readable program code for supplementing the PE using a PE
12 supplement, the PE supplement having a PE manifest representing the PE and a
13 PE identifier to identify the PE; and

14 computer readable program code for handling the PE by a PE handler using the
15 PK and the PE supplement.

1 42. The computer program product of claim 41 further comprises:
2 computer readable program code for booting up the platform by a boot-up code
3 following a power on.

1 43. The computer program product of claim 42 wherein the secure
2 environment includes an OSE supplement to supplement the OSE with an OSE
3 manifest representing the OSE and an OSE identifier to identify the OSE.

1 44. The computer program product of claim 43 wherein the computer
2 readable program code for handling the PE comprises:
3 computer readable program code for loading the PE and the PE supplement
4 from a PE memory into the isolated memory area using a parameter block provided by
5 the boot-up code;
6 computer readable program code for verifying the PE manifest; and
7 computer readable program code for verifying the PE using the PE manifest and
8 a constant derived from the PK.

1 45. The computer program product of claim 44 wherein the computer
2 readable program code for handling the PE further comprises:
3 computer readable program code for generating a PE key using the PK;
4 computer readable program code for logging the PE identifier in a storage; and
5 computer readable program code for handling a PE entry and a PE exit.

1 46. The computer program product of claim 45 wherein the computer
2 readable program code for generating the PE key comprises:
3 computer readable program code for combining the PE identifier and the , the
4 combined PE identifier and the PK corresponding to the PE key.

1 47. The computer program product of claim 46 wherein the computer
2 readable program code for handling the OSE comprises:

1 52. The computer program product of claim 51 wherein the computer
2 readable program code for managing the subset of the OS further comprises:
3 computer readable program code for generating an applet key associating with
4 the applet module.

[illegible]

1 53. The computer program product of claim 52 wherein the computer
2 readable program code for generating the applet key comprises:
3 computer readable program code for combining the OSE key with an applet
4 identifier identifying the applet module, the combined OSE key and the applet identifier
5 corresponding to the applet key.

1 54. The computer program product of claim 53 wherein the computer
2 readable program code for booting up comprises:
3 computer readable program code for locating the PE and the PE supplement;
4 computer readable program code for transferring the PE and the PE supplement
5 into the PE memory at a PE address;
6 computer readable program code for recording the PE address in the parameter
7 block; and
8 computer readable program code for executing an isolated create instruction, the
9 isolated create instruction loading the PE handler into the isolated memory area.

1 55. The computer program product of claim 54 wherein the computer
2 readable program code for executing the isolated create instruction comprises computer
3 readable program code for performing an atomic sequence, the atomic sequence being
4 non-interruptible.

1 56. The computer program product of claim 55 wherein the computer
2 readable program code for performing the atomic sequence comprises:
3 computer readable program code for verifying if the processor is in a flat
4 physical page mode;
5 computer readable program code for reading and incrementing a thread count
6 register in a chipset to determine if the processor is the first processor in the isolated
7 execution mode;
8 computer readable program code for configuring the chipset using a
9 configuration storage;
10 computer readable program code for configuring the processor in the isolated
11 execution mode; and

12 computer readable program code for loading the PE handler into the isolated
13 memory area.

1 57. The computer program product of claim 56 wherein the computer
2 readable program code for performing the atomic sequence further comprises:
3 computer readable program code for verifying the loaded PE handler; and
4 computer readable program code for transferring control to the loaded PE
5 handler.

1 58. The computer program product of claim 56 wherein the computer
2 readable program code for configuring the processor in the isolated execution mode
3 comprises:
4 computer readable program code for reading the configuration storage in the
5 chipset when the processor is not a first processor in the isolated execution mode; and
6 computer readable program code for configuring the processor according to the
7 configuration storage in the chipset when the processor is not the first processor in the
8 isolated execution mode.

1 59. The computer program product of claim 58 wherein the chipset includes
2 at least one of a memory controller hub (MCH) and an input/output controller hub
3 (ICH).

1 60. The computer program product of claim 48 wherein the storage is in an
2 input/output controller hub (ICH) external to the processor.

1 61. A system comprising:
2 a processor operating in one of a normal execution mode and an isolated
3 execution mode;
4 a memory coupled to the processor having an isolated memory area accessible
5 to the processor in the isolated execution mode; and
6 an executive subsystem comprising:
7 a processor executive (PE) to handle an operating system executive
8 (OSE) in a secure environment, the secure environment having a platform key

9 (PK) and associated with the isolated memory, the OSE to manage a subset of
10 an operating system (OS),
11 a PE supplement to supplement the PE with a PE manifest representing
12 the PE and a PE identifier to identify the PE, and
13 a PE handler to handle the PE using the PK and the PE supplement.

1 62. The system of claim 61 wherein the executive subsystem further
2 comprises:
3 a boot-up code to boot up the platform following a power on.

1 63. The system of claim 62 wherein the secure environment includes an
2 OSE supplement to supplement the OSE with an OSE manifest representing the OSE
3 and an OSE identifier to identify the OSE.

1 64. The system of claim 63 wherein the PE handler comprises:
2 a PE loader to load the PE and the PE supplement from a PE memory into the
3 isolated memory area using a parameter block provided by the boot-up code;
4 a PE manifest verifier to verify the PE manifest; and
5 a PE verifier to verify the PE using the PE manifest and a constant derived from
6 the PK.

1 65. The system of claim 64 wherein the PE handler further comprises:
2 a PE key generator to generate a PE key using the PK;
3 a PE identifier logger to log the PE identifier in a storage; and
4 a PE entrance/exit handler to handle a PE entry and a PE exit.

1 66. The system of claim 65 wherein the PE key generator comprises:
2 a PE key combiner to combine the PE identifier and the PK, the combined PE
3 identifier and the PK corresponding to the PE key.

1 67. The system of claim 66 wherein the PE comprises:
2 an OSE loader to load the OSE and the OSE supplement into the isolated
3 memory area;

4 an OSE manifest verifier to verify the OSE manifest; and
5 an OSE verifier to verify the OSE.

1 68. The system of claim 67 wherein the PE further comprises:
2 an OSE key generator to generate an OSE key;
3 an OSE identifier logger to log the OSE identifier in a storage; and
4 an OSE entrance/exit handler to handle an OSE entry and an OSE exit.

1 69. The system of claim 68 wherein the OSE key generator comprises:
2 a binding key generator to generate a binding key (BK) using the PE key; and
3 an OSE key combiner to combine the OSE identifier and the BK, the combined
4 OSE identifier and the BK corresponding to the OSE key.

1 70. The system of claim 69 wherein the OSE comprises:
2 a module loader to load a module into the isolated memory area;
3 a page manager to manage paging in the isolated memory area; and
4 an interface handler to handle interface with the OS.

1 71. The system of claim 69 wherein the module is one of an application
2 module, an applet module, and a support module.

1 72. The system of claim 71 wherein the OSE further comprises:
2 an applet key generator to generate an applet key associating with the applet
3 module.

1 73. The system of claim 72 wherein the applet key generator comprises:
2 an applet key combiner to combine the OSE key with an applet identifier
3 identifying the applet module, the combined OSE key and the applet identifier
4 corresponding to the applet key.

1 74. The system of claim 73 wherein the boot up code comprises:
2 a PE locator to locate the PE and the PE supplement, the PE locator transferring
3 the PE and the PE supplement into the PE memory at a PE address;

4 a PE recorder to record the PE address in the parameter block; and
5 an instruction invoker to execute an isolated create instruction, the isolated
6 create instruction loading the PE handler into the isolated memory area.

1 75. The system of claim 74 wherein the isolated create instruction performs
2 an atomic sequence, the atomic sequence being non-interruptible.

1 76. The system of claim 75 wherein the atomic sequence comprises:
2 a physical memory operation to verify if the processor is in a flat physical page
3 mode;
4 an atomic read-and-increment operation to read and increment a thread count
5 register in a chipset, the read-and-increment operation determining if the processor is
6 the first processor in the isolated execution mode;
7 an isolated memory area control operation to configure the chipset using a
8 configuration storage;
9 a processor isolated execution operation to configure the processor in the
10 isolated execution mode; and
11 an PE handler loading operation to load the PE handler into the isolated memory
12 area.

1 77. The system of claim 76 wherein the atomic sequence further comprises:
2 a PE handler verification to verify the loaded PE handler; and
3 an exit operation to transfer control to the loaded PE handler.

1 78. The system of claim 76 wherein the processor isolated execution
2 operation comprises:
3 a chipset read operation to read the configuration storage in the chipset when the
4 processor is not a first processor in the isolated execution mode; and
5 a processor configuration operation to configure the processor according to the
6 configuration storage in the chipset when the processor is not the first processor in the
7 isolated execution mode.

002260" 095858E 092200

1 79. The system of claim 78 wherein the chipset includes at least one of a
2 memory controller hub (MCH) and an input/output controller hub (ICH).

1 80. The system of claim 68 wherein the storage is in an input/output
2 controller hub (ICH) external to the processor.

002260" 58589960